Leveraging Ontologies upon a Holistic Privacy-aware Access Control Model

Eugenia I. Papagiannakopoulou[1], Maria N. Koukovini[1],
Georgios V. Lioudakis[1], Nikolaos Dellas[2], Joaquin Garcia-Alfaro[3],
Dimitra I. Kaklamani[1], Iakovos S. Venieris[1],
Nora Cuppens-Boulahia[4], and Frédéric Cuppens[4]

[1] School of Electrical and Computer Engineering,
National Technical University of Athens,
Heroon Polytechniou 9, 15773, Athens, Greece
[2] SingularLogic S.A., Al. Panagouli & Siniosoglou, 14234 N. Ionia, Greece
[3] Institut Mines-Telecom, Telecom SudParis,
CNRS Samovar UMR 5157, Evry, France
[4] Institut Mines-Telecom, Telecom Bretagne
CS 17607, 35576 Cesson-Sévigné, France

Abstract. Access control is a crucial concept in both ICT security and privacy, providing for the protection of system resources and personal data. The increasing complexity of nowadays systems has led to a vast family of solutions fostering comprehensive access control models, with the ability to capture a variety of parameters and to incorporate them in the decision making process. However, existing approaches are characterised by limitations regarding expressiveness. We present an approach that aims at overcoming such limitations. It is fully based on ontologies and grounded on a rich in semantics information model. The result is a privacy-aware solution that takes into consideration a variety of aspects and parameters, including attributes, context, dependencies between actions and entities participating therein, as well as separation and binding of duty constraints.

1 Introduction

In order to ensure ICT security and privacy, a given security policy must be defined. A security policy can be seen as a series of rules stating what is permitted and what is not permitted in a system during normal operations. Indeed, the policy must contain the complete set of requirements for the system in terms of security and data protection. This way, access control is the core component of any ICT system in terms of security and privacy protection.

Beyond legacy access control models, such as the well-adopted Role-Based Access Control (RBAC) [36], most of the prominent recent approaches typically propose enhancements of existing security models, and incorporate different criteria to take dynamic decisions. In that respect, access control models have incorporated concepts such as organisation [2], context [9], and attributes [42], among others, whereas, the consideration of features that are specific to privacy protection has resulted in the emergence of the field referred to as Privacy-Aware Access Control (cf. e.g., [3]). In this context, ontologies have also been proposed for the specification of complex access control policies.

This paper presents an access control approach that leverages the full potential of ontologies, in order to enable the specification of very expressive access control rules. The approach is grounded on an innovative access control model, developed in the frame of the FP7 ICT project DEMONS [11], and first presented in [29][30]. It aims at handling security and privacy requirements for distributed processes in a holistic and comprehensive manner. The proposed approach combines various features, including context, attributes, privacy-awareness, Separation and Binding of Duty [5], and a variety of dependencies, and has been successfully applied in the automatic privacy-aware verification and transformation of distributed workflows [23].

The rest of this paper is organised as follows. Section 2 surveys related work. Section 3 presents the Information and Policy models underlying our solution. Sections 4 and 5 present, respectively, the ontological implementation of such models. Section 6 describes an extension providing support for *offline* knowledge extraction. Finally, Section 7 concludes the paper.

2 Related Work

The advent of the Semantic Web and the technologies it brings, especially semantic ontologies, have provided access control with new potentials. Therefore, several approaches have leveraged Semantic Web technologies in various ways, seeking expressiveness, formal semantics and reasoning capabilities; as a starting point, the Web Ontology Language (OWL) [39] was used to develop policy languages for the Web, such as Rei and KAoS [41], as well as to provide interoperability while accessing heterogeneous databases, as in [26][28][38].

Since RBAC [36] constitutes the baseline for access control, various approaches targeting its ontological implementation have been proposed. In this context, ontologies are used to represent the main concepts of RBAC —Action, Subject, Object, Role, Permission— as well as role hierarchies and dynamic and static Separation of Duty (SoD) constraints. An important work in this field is presented in [15], where R*OWL*BAC is introduced, proposing two different approaches regarding role representation: the first maps roles to classes and subclasses to which individual subjects can belong, whereas the second represents roles as instances of the generic `Role` class. Similarly, the approach referred to as XACML+OWL [14] combines OWL with XACML [27], with a view to decouple the management of constraints and RBAC hierarchies from the specification and the enforcement of the actual XACML policies. On the other hand, approaches such as [8][18] combine RBAC with the Attribute Based Access control (ABAC) paradigm [42], in order to take into account attributes during the definition of policies and the access control decision.

Apart from XACML+OWL [14], several approaches have leveraged XACML together with ontologies, most of them targeting the expression limitations of the attribute-based paradigm. In this direction, the approach described in [32] proposes an ontology-based inference engine which extends XACML attribute management for simplifying the specification and maintenance of ABAC policies. The work presented in [20] addresses the expressiveness limitations of XACML regarding knowledge representation; it extends it in order to support ontology-based reasoning and rule-based inference, while maintaining the usability of its original features. Likewise, in [34] an XML filter is created for regulating the disclosure of information, according to both the XML document structure and the semantics of its contents; this is achieved by directly integrating a knowledge base, which contains a description of the domain, in an XACML engine.

An important aspect of access control is reflected by the concept of *context*, which generally refers to information describing a specific situation; context includes static and dynamic environmental characteristics, such as temporal, spatial and historical ones. In that respect, there have been proposed various extensions to well established models in order to include contextual constraints, such as the Extended RBAC Profile of XACML, presented in [1]. A prominent approach in this area constitutes the Temporal Semantic Based Access Control (TSBAC) model [33], which enhances the specification of user-defined authorisation rules by constraining time interval and temporal expressions over users' history of accesses, which are stored in a History Base. OrBAC [2][9][31] is rather the most mature approach in this area; it is the first to express all different types of context within a unique homogeneous framework. In particular, OrBAC defines a *Context Ontology* comprised not only of temporal, spatial, and historical context but also of *user-declared* and *application dependent* context; the latter depends on the characteristics that join the subject, the action and the object and can be evaluated by querying the system database, whereas user-declared context allows for modelling contexts that are difficult to be described using environmental conditions.

The complex relations considered in Online Social Networks (OSNs) and the associated applications highlight the need for semantic organisation of the contained knowledge and for semantic access control mechanisms. In this context, the work presented in [12] leverages ontologies for representing relationships with the individuals and the community in order to determine the access restrictions to community resources. Carminati et al. provide in [6] a much richer OWL ontology for modelling various aspects of OSNs, while also proposing *authorisation*, *administration* and *filtering* policies that depend on trust relationships among various users. A more detailed approach is presented in [25], which proposes the Ontology-based Social Network Access Control (OSNAC) model, encompassing two ontologies; the Social Networking systems Ontology (SNO), capturing the information semantics of a social network, and the Access Control Ontology (ACO), which allows for expressing access control rules on the relations among concepts in the SNO.

However, all the approaches described above present limitations as far as *expressiveness* is concerned. In most cases, they focus on and capture a limited number of concepts, constraints and access parameters, missing the necessary expressiveness for the specification of complex provisions and access structures. For instance, only few of these models (e.g., [6][12][25][34]) are privacy-aware, yet they do not provide support for separation and binding of duty constraints, whereas presenting limited, if any, context-awareness. Moreover, the semantic taxonomies created within existing approaches are typically limited to very basic hierarchies (e.g., of roles), with no support for relations beyond *is-a* generalisations, or complex expressions and logical relations thereof. These limitations have been the motivation for the development of a new model, being *holistic* in terms of providing the means for incorporating a manifold of concepts and features, as described in the following sections.

3 Policy-based Access Control Model

This Section outlines the policy-based access control model, on which the proposed ontological approach has been based. The starting point for this work has been the data protection legislation and related policy-oriented best practice guidelines, which provide, and often codify, the fundamental principles surrounding the provision of privacy-aware services. These typically concern lawfulness of data collection and processing, purpose specification and binding, necessity, adequacy, proportionality and quality of the data processed, minimal use of personal information, application of security measures, special provisions regarding retention and protection of information, enforcement of data subjects rights, coordination with the competent authorities, etc. The elaboration of principles and requirements stemming from the legislation and fair information practices have been the subject of various studies and extensive research (e.g., [16][17][24][37]). Rethought from the point of view of access control, the corresponding principles converge to the following challenges:

Multi-aspect access rights definition: Given the inherent complexity of the notion of privacy and the underlying implications, the associated solutions should incorporate various criteria in access and usage control decisions, rather than just *which user* holding *which role* is performing *which action* on *which object*.

Purpose: The "purpose principle" is essential for privacy awareness, being a core part of data collection and processing lawfulness [13]; a privacy-aware access control framework should provide for purpose specification and binding.

Privacy-aware information flow: Beyond controlling access and usage, a privacy-aware access control model should provide for the specification of acceptable patterns as far as the flow of data is concerned; this implies, for instance, the prevention of some data to be communicated from a system to another, whereas the latter may be *per se* allowed to receive the same data by a third system.

Unlinkability: Along the same line, a privacy-aware access control model should provide support for preventing linkability. Whereas privacy-aware information flow refers to "direct" passing of data among systems, processes or people, the need for unlinkability reflects a generalization towards mutually exclusive availability or processing of data, either explicitly or implicitly.

Separation and Binding of Duty (SoD/BoD): Similarly, SoD and BoD constraints should be possible to be specified and enforced, since they hold an important position among authorization requirements [21], serving, among others, conflicts avoidance and unlinkability.

Complementary actions: In several cases, access to the data should be accompanied by certain actions that should follow the collection and/or processing of information. These are often referred to in the literature as "privacy obligations" ([7][19]) and may concern, for instance, the application of immediate protection measures, the interaction with the data subjects (e.g., in terms of information or request for consent), and the enforcement of data retention provisions.

Context-awareness: It has become apparent that effective security and privacy policies largely depend on contextual parameters ([9][22]). Therefore, a privacy-aware access control framework should incorporate the corresponding aspects, in terms of restrictions over contextual parameters and events, and be enabled to impose different access rights according to the applicable constraints.

Semantics: Vertical to all the above is the need for precise semantics of the underlying concepts; data, actors, actions, context, purposes, among others, should be semantically defined, fostering transparency, accountability and effectiveness in terms of privacy.

The policy-based access control model presented here has been specified according to and achieves to address all the highlighted requirements for privacy awareness in access control.

3.1 Information Model

The day-to-day operation of an organisation involves a variety of entities, like machines, users and data[5]. We consider two representation levels; the *concrete level* refers to well-specified entities, e.g., named humans, while the *abstract level* enables referring to entities by using abstractions, especially their semantic type and attributes. The main concepts considered by the model are presented in Table 1.

At a concrete level, the set of *Users* (U) represents human entities, while this of *Organisations* (Org) describes internal divisions (e.g., departments) or external parties (e.g., sub-contractors). The various machinery comprise the *Machines* (M) set, providing hosting to *Operation Containers* (OpC) that offer *Operation Instances* (OpI). Operation Instances correspond to actual implementations of functionalities, while Operation Containers bundle collections of Operation Instances provided by the same functional unit[6]. Finally, information comprises the set of *Data* (D).

[5] Naturally, the information model may vary depending on the application domain; still, several concepts (e.g., organisational roles, operations, data types, etc.) are pervasive and are the focus of the following.

[6] In Web Services terms, Operation Containers correspond to a service *interface*, whereas Operation Instances represent the associated *operations* [40].

All above elements constitute instantiations of their semantic equivalents described at the abstract level. Users are assigned with *Roles* (R), Operation Instances provide implementations of *Operations* (Op), while data, organisations, machines and operation containers have *types*, reflecting the semantic class they fall under; thus, sets of *Data Types* (DT), *Organisation Types* ($OrgT$), *Machine Types* (MT) and *Operation Container Types* ($OpCT$) are defined. The semantic model also includes *Context Types* ($ConT$), enabling the definition of contextual parameters, *Attributes* (Att), leveraged for describing properties and characteristics of other elements, and *Purposes* (Pu) justifying access requests.

Table 1. Concepts of the Information Model

Abstract Level	**Concrete Level**	**Description**	***Act***	***Res***
Data Types (DT)	Data (D)	Data being collected and/or processed, organised according to their semantic types		✓
Roles (R)	Users (U)	Human users assigned with roles reflecting their responsibilities inside an organisation	✓	✓
Operations (Op)	Operation Instances (OpI)	Operations reflect all actions that can take place in the context of the system's operation	✓	✓
Operation Container Types ($OpCT$)	Operation Containers (OpC)	Components or other functional structures that typically offer a set of operations together	✓	✓
Machine Types (MT)	Machines (M)	Hardware components hosting operation containers		✓
Organisation Types ($OrgT$)	Organisations (Org)	The various domains within which actions are performed		
Context Types ($ConT$)	Context values	Real-time parameters and events		
Purposes (Pu)	(No concrete representation)	Purposes for which access to resources is requested		
Attributes (Att)	Attribute values	Characteristics further describing members of the other sets		

All concepts shown in Table 1 comprise graphs of elements characterised by relations; the latter are implemented by predicates defining AND- and OR-hierarchies and enable the inheritance of attributes and rules, as well as the specification of dependencies. For instance, and with respect to the DT graph, three partial order relations are defined: $isA(dt_i,\ dt_j)$, $lessDetailedThan(dt_i,\ dt_j)$ and $isPartOf(dt_i,\ dt_j)$, where $dt_i,\ dt_j \in DT$, reflecting the particularisation of a concept, the detail level and the inclusion of some data types to another, respectively. Moreover, the model specifies the necessary predicates in order to link concepts from different graphs; for example, the predicate $mayActForPurposes(r,\ \langle pu \rangle^k)$, where $r \in R$, $\langle pu \rangle^k \subseteq \mathcal{P}(Pu)$, indicates the legitimate purposes $\langle pu \rangle^k$ for which the users assigned with the role r may act.

3.2 Actions

The entities of the Information Model participate in the definition of *Actions* (*Act*), that are the main components of access control rules. An *action* refers to the situation where an *actor* performs an *operation* on a *resource*. Different types of entities may play the role of actors and resources, thus be members of the corresponding *Actors* (A) and *Resources* (*Res*) sets, as indicated in Table 1. An action is defined as follows.

Definition 1. An *action* $act_i \in Act$ is a tuple $\langle a_i,\ op_i,\ res_i,\ org\rangle$, such that: $act_i \in A$ is an actor; $op_i \in Op$ is an operation; $res_i \in Res$ is a resource; and $org \in Org$ is the organisation within which an action takes place.

An action can be either *atomic* or *composite*, depending on whether the associated operation can be decomposed to more elementary operations or not, following the hierarchical relations in *Op*. Actions are also categorised to *abstract*, *concrete* and *semi-abstract*, depending on whether actors and resources are defined at abstract, concrete or mixed level.

Finally, it should be stressed that the elements of an action can be specified as *enhanced entities* that include, apart from the entity's semantic type, expressions over its attributes and/or sub-concepts, thus refining the concept definition, towards specifying attribute-based constraints and access control rules.

3.3 Access Control Rules

Access control rules are used for defining *permissions*, *prohibitions* and *obligations* over actions and, since actions can be abstract, concrete or semi-abstract, rules are also specified at these three levels. They are defined as follows.

Definition 2. An *access control rule* is a structure:

$$\left.\begin{array}{r} Permission \\ Prohibition \\ Obligation \end{array}\right\} (pu,\ act,\ preAct,\ cont,\ postAct)$$

where $act \in Act$ is the action that the rule applies to; $pu \in Pu$ is the purpose for which *act* is permitted/prohibited/obliged to be executed; $cont \in \mathcal{P}(ConT)$ is a structure of contextual parameters; $preAct \in Act$ is a structure of actions that should have preceded; $postAct \in Act$ refers to the action(s) that must be executed following the rule enforcement.

An important observation here is that the concept of organisation is not involved in the rules' body, but instead it is specified for each action; although a rule concerns the execution of an action within an organisation, pre- and post-actions may take place within other organisations.

Apart from single actions, pre- and post- actions may also refer to structures of actions. Thus, they may consist of actions interrelated by means of logical operators $\wedge$ and $\vee$, including negation, i.e., $\neg preAct$, $\neg postAct$. The term *Skeleton* is used to denote structures of actions following various sequence patterns. In addition, pre-/post-actions may be characterised by *sequence constraints*, putting constraints regarding *when* they are executed with respect to the action that the rule applies to.

4 Information Model Ontology

Fig. 1 provides an overview of the Information Model Ontology (IMO). As shown, all abstract concepts described in §3.1 and summarised in Table 1 comprise classes, characterised by intra- and inter-class relations that are implemented as OWL object properties. The main intra-class properties are `isA`, `isPartOf` and `moreDetailedThan` that, along with their inverses[7], essentially comprise AND- and OR- hierarchies, enabling inheritance, as well as dependencies specification. Inter-class relations describe associations between concepts of different classes, indicating, for instance, the roles that may act for a purpose (`mayActForPurpose`), or the attributes characterising a concept (`hasAttribute`).

Individuals of the `Attributes` class are associated with an identifier (`AttributeNames`), a type, that can be a usual type (e.g., "Integer") or an IMO entity, and optionally a value, which can be an ontological element, or an arbitrary string, declared using the `hasValue` and `hasStringValue` properties, respectively. A valued attribute is considered *immutable*, as opposed to *mutable* attributes, the values of which are free to be determined during execution. Finally, instances of the `DataIO` class map an operation with its inputs and outputs, indicating the attributes characterising each input/output relation, as well as the associated `States`, referring to different states of information, such as "anonymised" vs. "identifiable". States are very important for applying access control at large scale, in the context of workflow verification [23].

5 Policy Model Ontology

Fig. 2 provides an overview of the Policy Model Ontology (PMO), while its main aspects are described in what follows. Further, Fig. 3 illustrates the ontological representation of an example rule, inspired from guidelines for the health sector [10]: *"For the purpose of medical research and in the context of an ongoing R&D project, a statistician is allowed to perform statistical analysis on identifiable medical records of a patient, if the said patient has provided consent therefor; for accountability reasons, access should be immediately logged"*.

5.1 Expressions and Logical Relations

In the direction of achieving rich expressiveness, two useful tools are *expressions* and *logical relations*. The latter allow specifying logical structures of con-

[7] Inverse properties are explicitly defined for all object properties in the ontology, in order to ease navigation from one ontological element to another.

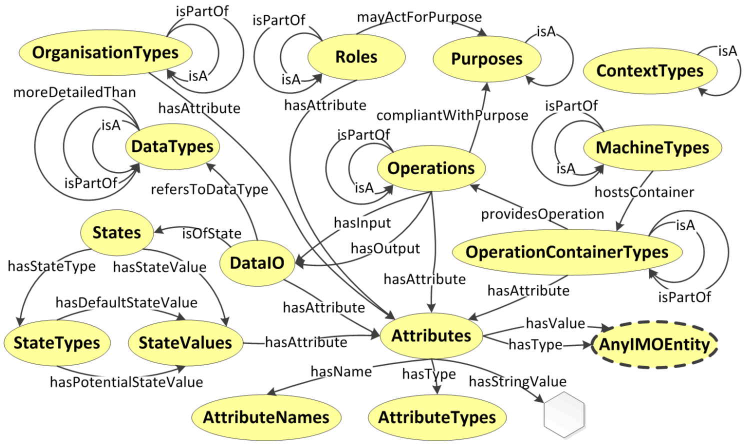

Fig. 1. Information Model Ontology (IMO).

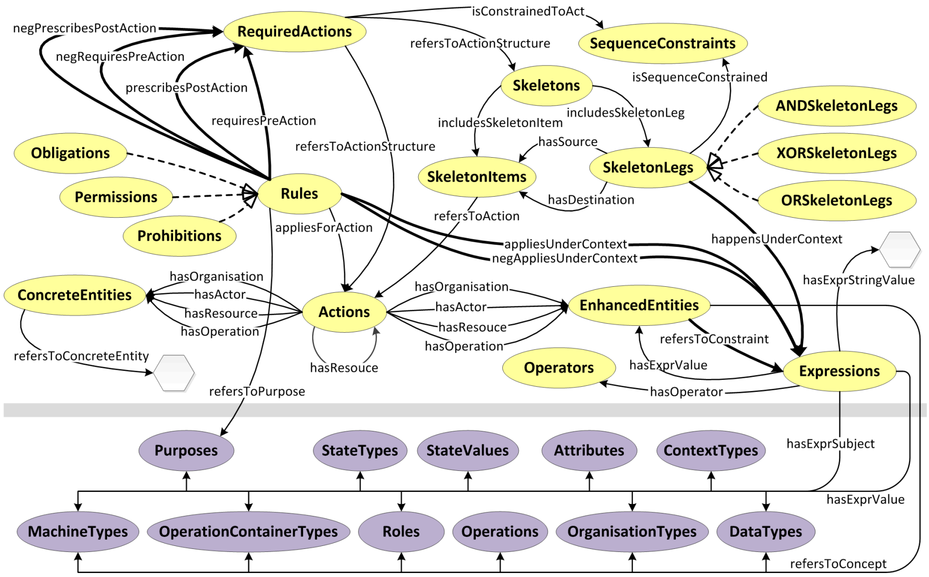

Fig. 2. Policy Model Ontology (PMO).

cepts. For instance, a rule may specify different post-actions to be jointly executed (AND), or pre-actions that should precede inclusively (OR) or exclusively (XOR). A logical relation is defined as follows.

Definition 3. Let $\mathcal{F}$ be the class of all functions on a set S, such that each $\phi_i(V) \in \mathcal{F}$ is a well-formed formula built up from the n-ary operators AND, OR and XOR, the unary operator NOT, and a set V of variables; a *logical relation* is a logical structure $\phi(S')$, such that $\phi \in \mathcal{F}$ and $S' \subseteq S$.

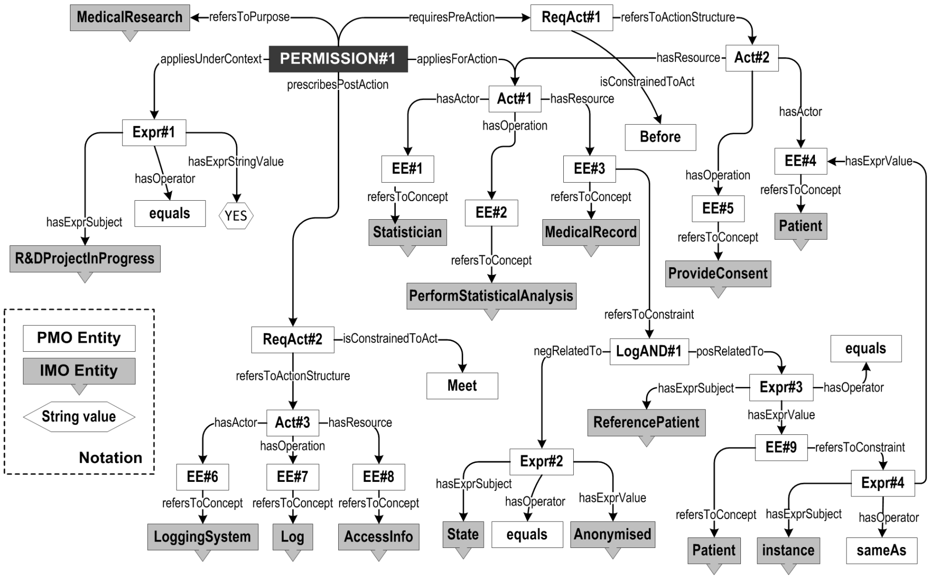

Fig. 3. Example of Ontological Access Control Rule.

Thick lines in Fig. 2 imply the use of logical relations for structuring PMO elements; they are implemented by means of the `LogicalRelations` class (not shown in Fig. 2). Instances of its subclasses `ANDRelations`, `ORRelations`, `XORRelations` represent the AND, OR and XOR operators. Ontological instances participating in logical relations, including other logical relations, are referenced through the `posRelatedTo` and `negRelatedTo` properties, with the latter modelling the use of the NOT operator.

Expressions enable the definition of contextual conditions and constraints on concepts (e.g., on an actor's attributes); they comprise ternary relations assigning a value to a subject through an operator, or logical structures of such triples.

Definition 4. An *atomic expression* is a tuple $\langle exprSubject, operator, exprValue\rangle$, such that: *exprSubject* reflects the reference concept; $operator \in Operators$, the latter being a set of operators, such as `equals`, `greaterThan`, etc.; *exprValue* represents the value assigned to the *exprSubject*. An *expression* is either an atomic expression or a logical relation thereof.

Ontologically, expressions are modelled by means of the `Expressions` and `LogicalRelations` classes; instances of the former essentially model atomic expressions, whereas the latter provide for structuring composite expressions out of atomic ones. Based on Definition 4, appropriate properties are defined for `Expressions` individuals, indicating the subject (`hasExprSubject`), operator (`hasOperator`) and value (`hasExprValue`). Operators are ontologically defined as individuals of the `Operators` class, while the subject and object of an expression can be individuals of both PMO and IMO. However, an expression value can also be arbitrary, i.e., not an ontologically defined concept; in such case, the hasExprStringValue datatype property is used instead of hasExprValue, in order to assign a String value.

5.2 Actions and Entities

As stressed in Section 3, actions lie at the core of access control rules; not only rules are applied over actions, but they also appoint pre- and post-actions that should (or not) be executed before and after a rule's enforcement. Ontologically, actions are implemented as Actions class instances. Following Definition 1, $\langle a_i$, op_i, res_i, $org\rangle$ is reproduced by means of object properties indicating the actor (hasActor), operation (hasOperation), resource (hasResource) and organisation (hasOrganisation) of the action (Fig. 2).

However, the afore-described properties do not point directly to the reference concept, such as a role declared as the actor; instead, the proposed approach makes use of intermediate objects, being instances of the EnhancedEntities class. An enhanced entity not only indicates the reference abstract entity described in IMO, but it also defines constraints for this entity, thus enabling the enforcement of attribute-based access control. The corresponding object properties are refersToConcept and refersToConstraint, with the latter pointing at either an Expressions or a LogicalRelations instance, whereas the constraints are defined on the attributes of the entity, or its elements, i.e., individuals related (directly or indirectly) through the isPartOf object property.

As mentioned in §3.2, actions may contain elements defined at the concrete level; therefore, the class ConcreteEntities is defined for representing such entities, e.g., a specific user, instead of abstract concepts, by means of the refersToConcreteEntity datatype property.

Fig. 3 illustrates three actions, corresponding to the statistical analysis (Act#1), the pre-action of consent provision (Act#2), and the post-action of logging (Act#3). These involve various enhanced entities, most of which are unconstrained, such as EE#1 corresponding to the Statistician actor, or EE#2 reflecting the statistical analysis operation. On the other hand, EE#3 referring to the MedicalRecord resource has two constraints, described by expressions Expr#2 and Expr#3, and associated through an AND logical relation (LogAND#1). Specifically, Expr#2 implies identifiable data, through negation over Anonymised state, while Expr#3 is an example of concepts' binding, further elaborated in §5.5.

Finally, it is important to note that actions themselves may comprise resources of other actions. This is the case with Act#1, comprising the resource of Act#2, in the sense that the patient must have provided consent for Act#1 execution.

5.3 Ontological Access Control Rules

Access control rules are implemented in PMO by means of the Rules class (Fig. 2); as rules may describe *permissions*, *prohibitions*, or *obligations*, this is appropriately sub-classed by Permissions, Prohibitions and Obligations. Each rule is described by a Rules instance defining its elements, that is, the action it applies for, the pre-/post-actions, the contextual conditions and the underlying purpose.

In this context, refersToPurpose property maps a rule with a Purposes instance, whereas appliesUnderContext and its negative equivalent negAppliesUnderContext point at an Expressions or LogicalRelations (having Expressions as leaves) instance, declaring the contextual parameters under which the considered rule applies. For example, the permission of Fig. 3 applies for the purpose of MedicalResearch, given that an R&D project is in progress (R&D-ProjectInProgress).

The main action of the rule is an Actions instance, directly defined through the appliesForAction property. As for pre- and post-actions, they are also Actions instances; however, the rule is connected with RequiredActions instances that mediate between the rule and the actions, through the requiresPreAction, prescribesPostAction, and their negative variants. This choice is motivated by two introduced features: first, it enables the description of complex actions structures, referred to as *skeletons* (cf. §5.4); second, it allows putting constraints regarding *when* a pre-/post-action is executed with respect to the main action of the rule. In this context, the isConstrainedToAct property is leveraged for expressing temporal and sequence constraints, expressed by instances of the SequenceConstraints class. In the example, the use of Meet imposes a strict temporal constraint, prescribing that the end of the main action should coincide with the beginning of the post-action (Act#3), whereas Before implies a loose sequence constraint, meaning that the pre-action Act#2 should be executed sometime before the main action.

5.4 Skeletons

In order to enable combination of actions so as to form complex structures thereof, the concept of *skeletons* is introduced. Implemented as instances of the Skeletons class, they provide the means for the definition of actions' structures, together with their sequential associations. Skeletons can comprise pre- and post- actions, being referred to by RequiredActions instances by means of the refersToActionStructure property.

The underlying actions are indicated by instances of the SkeletonItems class through the refersToAction property, whereas SkeletonLegs describe the interaction patterns among them. A skeleton leg is essentially an *edge* connecting two actions; it has an initial and a terminal skeleton item, appointed by hasSource and hasDestination properties, while it can be subject to contextual conditions, as well as to sequence constraints. The latter are implemented by means of SequenceConstraints, describing also whether the leg is *critical* or *non-critical* regarding the potential intervention of other actions between the initial and terminal skeleton items. Finally, for more flexibility in describing whether the implied transition will occur or not, three SkeletonLegs subclasses reflect, respectively, AND, OR and XOR associations among an action's outbound legs.

5.5 Separation and Binding of Duty

The high expressiveness of the proposed approach enables the specification of advanced Separation and Binding of Duty (SoD/BoD) constraints. Instead of relying on role-/user- centric constraints, it allows for SoD/BoD application to all elements comprising an action, i.e., the actor, the operation, the resource and the organisation. This is achieved by dependencies among the entities comprising the actions of a rule.

For instance, consider the case of the MedicalRecord, being the resource of statistical analysis (Act#1) in Fig. 3; it is assumed to contain the ReferencePatient field, i.e., $\texttt{ReferencePatient}\xrightarrow{\texttt{isPartOf}}\texttt{MedicalRecord}$, indicating the patient it refers to. Since Patient is a Roles instance, it has to be explicit that it is not *any* patient who has provided consent, but the one being the data subject of the MedicalRecord. In that respect, EE#9 is constrained by Expr#4, specifying that the reference patient instance should be sameAs the patient implied by EE#4.

6 Offline Reasoning over Access Control Rules

The core Policy Model Ontology described so far is extended in order to support *Offline Reasoning*, i.e., proactive extraction of knowledge contained in the access control rules. Through the Offline Reasoning procedure, all the required knowledge becomes available already by the request time, thus reducing the number of queries to the ontology and offering performance gains. In other words, all the heavy processing tasks are performed offline and only when the PMO is updated, for instance, when new access control rules are added or existing ones are revoked.

For this purpose, and as illustrated in Fig. 4, two classes have been specified, namely OfflineReasoningActions and OfflineRequiredActions. The instances contained in the first class represent all the actions permitted to be executed in the context of the system's operation; in that respect, OfflineReasoningActions instances are derived by the specified Permissions instances. An offline reasoning action refers to the original action being the access action of the considered permission; it is valid under a purpose and some contextual conditions and requires or forbids the presence of other action structures.

Obviously, OfflineRequiredActions class reflects the required or forbidden pre- and post-actions complementing the considered permitted access action. As opposed to the OfflineReasoningActions, instances of this class are derived not only from Permissions, but from any kind of rules. Essentially, while constructing an offline reasoning action, for each original access action all the permissions, prohibitions and obligations by which this action is referred, either as the main action of the rule or as a pre-/post-action, are gathered. In the

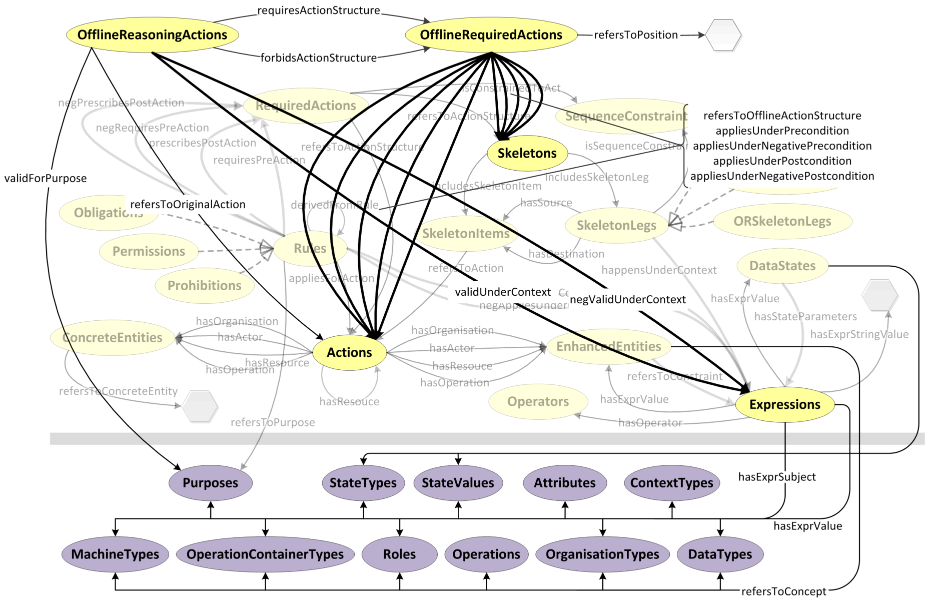

Fig. 4. Offline Reasoning in the PMO.

case of a permission, the specified pre- and post-actions are directly mapped to offline required actions, with the offline reasoning action referring to the main action of the rule, while in the case of prohibitions and obligations, we search for those where the considered offline reasoning action participates in the pre-action structure of the rule. Interrelations among pre- and post-actions of the found rules prescribe the need to also define pre- and post-conditions on the offline required actions.

Fig. 5 shows the offline reasoning action derived from the permission of Fig. 3. A question about the validity of the action ⟨Statistician, PerformStatisticalAnalysis, MedicalRecord⟩ would return the valid purpose MedicalResearch, the valid context Expr#1, as well as the offline required action structures OffReqActStr#1 and OffReqActStr#2.

7 Conclusion

This paper has presented an innovative policy-based privacy-aware access control model, focusing on its ontological implementation. Due to the high complexity and expressiveness of the base approach, we have chosen to implement it as an OWL ontology; this presents by itself various advantages, including formal and machine interpretable semantics, semantic consistency, inference of knowledge not explicitly contained in the ontologies, as well as direct integration with the DEMONS workflow management system, relying on comprehensive workflows also specified by means of ontologies. In fact, a success story behind the described

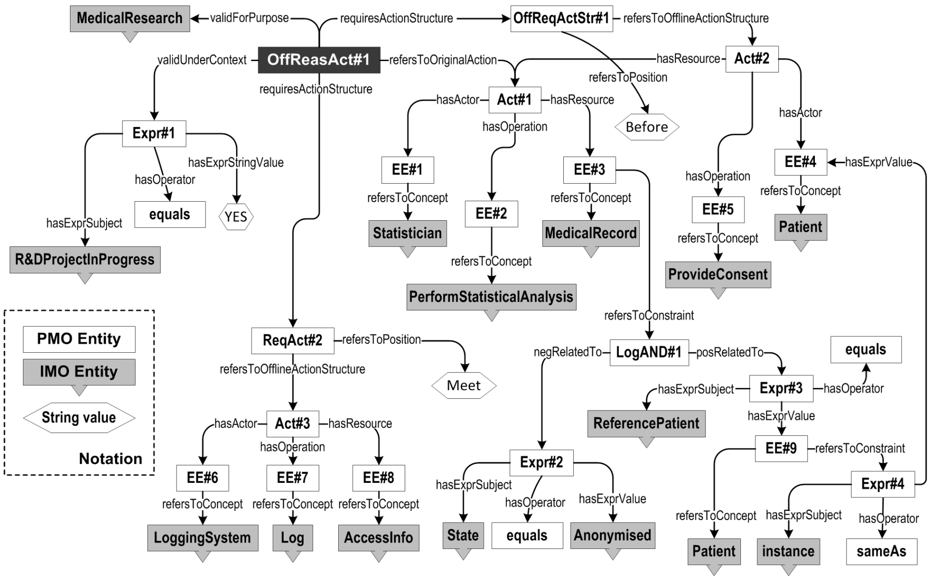

Fig. 5. Example of Offline Reasoning Action.

model has been its use in the context of privacy-aware workflow verification and transformation.

The proposed framework relies on policies that are built on top of a rich information model, implemented as an ontology, while the associated rules are specified over *actions* that reflect operational activities and can be described in different abstraction levels. The major advantage of the approach is its *expressiveness*, combining a manifold of advanced features; these include attributed entities and constraints, context awareness, the specification of complex dependencies among actions and entities, as well as sophisticated SoD and BoD provisions.

Perspectives for future work mainly concern aspects pertaining to the enforcement of the underlying provisions. In particular, a research priority is to combine the proposed model with the Attribute-Based Encryption (ABE) paradigm [4][35], in order to make possible to evaluate access rights by cryptographic means. In this context, the mechanisms for deriving access policies implemented with ABE from the proposed model are under investigation, extending the scope of offline knowledge extraction.

Acknowledgment

The research of M. N. Koukovini is co-financed by the European Union (European Social Fund — ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference

Framework (NSRF) — Research Funding Program: *Heracleitus II. Investing in knowledge society through the European Social Fund.* This research was also supported by the European Commission, in the frame of the FP7 DEMONS project.

References

1. Abi Haidar, D., Cuppens-Boulahia, N., Cuppens, F., Debar, H.: An extended RBAC profile of XACML. In: Proceedings of the 3rd ACM workshop on Secure web services. pp. 13–22. SWS '06, ACM, New York, NY, USA (2006)
2. Abou-El-Kalam, A., Baida, R.E., Balbiani, P., Benferhat, S., Cuppens, F., Deswarte, Y., Miège, A., Saurel, C., Trouessin, G.: Organization Based Access Control. In: 4th IEEE International Workshop on Policies for Distributed Systems and Networks (Policy'03). pp. 120–131 (June 2003), lake Come, Italy
3. Antonakopoulou, A., Lioudakis, G.V., Gogoulos, F., Kaklamani, D.I., Venieris, I.S.: Leveraging access control for privacy protection: A survey. In: Yee, G. (ed.) Privacy Protection Measures and Technologies in Business Organizations: Aspects and Standards, pp. 65–94. IGI Global (2012)
4. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: Proceedings of the 2007 IEEE Symposium on Security and Privacy. pp. 321–334. SP '07, IEEE Computer Society, Washington, DC, USA (2007)
5. Botha, R.A., Eloff, J.H.P.: Separation of duties for access control enforcement in workflow environments. IBM Systems Journal 40(3), 666–682 (2001)
6. Carminati, B., Ferrari, E., Heatherly, R., Kantarcioglu, M., Thuraisingham, B.: A semantic web based framework for social network access control. In: SACMAT '09: Proceedings of the 14th ACM symposium on Access control models and technologies. pp. 177–186. ACM (2009)
7. Casassa Mont, M.: Dealing with privacy obligations: Important aspects and technical approaches. In: Katsikas, S., Lopez, J., Pernul, G. (eds.) Trust and Privacy in Digital Business, pp. 120–131. Lecture Notes in Computer Science, Springer Berlin Heidelberg (2004)
8. Cruz, I.F., Gjomemo, R., Lin, B., Orsini, M.: A constraint and attribute based security framework for dynamic role assignment in collaborative environments. In: Bertino, E., Joshi, J.B.D. (eds.) CollaborateCom 2008: Collaborative Computing: Networking, Applications and Worksharing. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 10, pp. 322–339. Springer (2008)
9. Cuppens, F., Cuppens-Boulahia, N.: Modeling Contextual Security Policies. International Journal of Information Security 7(4), 285–305 (2008)
10. Data Protection Commissioner of Ireland: Data Protection Guidelines on research in the Health Sector (2007)
11. DEMONS (DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS) EU FP7 project: `http://fp7-demons.eu/`
12. Elahi, N., Chowdhury, M., Noll, J.: Semantic Access Control in Web Based Communities. In: ICCGI 2008: Proceedings of the Third International Multi-Conference on Computing in the Global Information Technology. pp. 131–136. IEEE Computer Society (August 2008)
13. European Parliament and Council: Directive 95/46/EC of the European Parliament and of the Council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. Official Journal of the European Communities L 281, 31–50 (November 1995)

14. Ferrini, R., Bertino, E.: Supporting rbac with xacml+owl. In: SACMAT '09: Proceedings of the 14th ACM Symposium on Access Control Models and Technologies. pp. 145–154. ACM (2009)
15. Finin, T.W., Joshi, A., Kagal, L., Niu, J., Sandhu, R.S., Winsborough, W.H., Thuraisingham, B.M.: ROWLBAC: representing role based access control in OWL. In: SACMAT '08: Proceedings of the 13th ACM Symposium on Access Control Models and Technologies. ACM (2008)
16. Gutwirth, S., De Hert, P., Poullet, Y.: Reinventing Data Protection? Springer, Berlin (2009)
17. Gutwirth, S., De Hert, P., Poullet, Y.: European Data Protection: Coming of Age. Springer, Berlin (2013)
18. He, Z., Huang, K., Wu, L., Li, H., Lai, H.: Using semantic web techniques to implement access control for web service. In: Zhu, R., Zhang, Y., Liu, B., Liu, C. (eds.) Information Computing and Applications, Communications in Computer and Information Science, vol. 105, pp. 258–266. Springer Berlin Heidelberg (2011)
19. Hilty, M., Basin, D., Pretschner, A.: On obligations. In: Vimercati, S., Syverson, P., Gollmann, D. (eds.) Computer Security ESORICS 2005, pp. 98–117. Lecture Notes in Computer Science, Springer Berlin Heidelberg (2005)
20. I. Ching Hsu: Extensible access control markup language integrated with semantic web technologies. Information Sciences 238, 33 – 51 (2013)
21. Joshi, J.B.D., Shafiq, B., Ghafoor, A., Bertino, E.: Dependencies and separation of duty constraints in GTRBAC. In: SACMAT '03: Proceedings of the 8th ACM Symposium on Access Control Models and Technologies. ACM (2003)
22. Kapitsaki, G.M., Lioudakis, G.V., Kaklamani, D.I., Venieris, I.S.: Privacy Protection in Context-Aware Web Services: Challenges and Solutions. In: Quan Z. Sheng, Jian Yu, S.D. (ed.) Enabling Context-Aware Web Services: Methods, Architectures, and Technologies, pp. 393–420. Chapman and Hall/CRC (2010)
23. Koukovini, M.N., Papagiannakopoulou, E.I., Lioudakis, G.V., Kaklamani, D.I., Venieris, I.S.: A workflow checking approach for inherent privacy awareness in network monitoring. In: DPM 2011: Proceedings of the 6th International Workshop on Data Privacy Management. LNCS, vol. 7122. Springer (2011)
24. Lioudakis, G.V., Gaudino, F., Boschi, E., Bianchi, G., Kaklamani, D.I., Venieris, I.S.: Legislation-aware privacy protection in passive network monitoring. In: Portela, I.M., Cruz-Cunha, M.M. (eds.) Information Communication Technology Law, Protection and Access Rights: Global Approaches and Issues, chap. 22, pp. 363–383. IGI Global (2010)
25. Masoumzadeh, A., Joshi, J.: OSNAC: An Ontology-based Access Control Model for Social Networking Systems. In: Proceedings of the 2010 IEEE Second International Conference on Social Computing. pp. 751–759. SOCIALCOM '10, IEEE Computer Society, Washington, DC, USA (2010)
26. Mitra, P., Pan, C.C., Liu, P., Atluri, V.: Privacy-preserving semantic interoperation and access control of heterogeneous databases. In: ASIACCS '06: Proceedings of the 2006 ACM Symposium on Information, computer and communications security. pp. 66–77. ACM (2006)
27. Organization for the Advancement of Structured Information Standards (OASIS): eXtensible Access Control Markup Language (XACML) Version 2.0. `http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf` (February 2005), OASIS Standard
28. Pan, C.C., Mitra, P., Liu, P.: Semantic access control for information interoperation. In: SACMAT '06: Proceedings of the 11th ACM symposium on Access control models and technologies. pp. 237–246. ACM, New York, NY, USA (2006)

29. Papagiannakopoulou, E.I., Koukovini, M.N., Lioudakis, G.V., Garcia-Alfaro, J., Kaklamani, D.I., Venieris, I.S.: A contextual privacy-aware access control model for network monitoring workflows: Work in progress. In: Garcia-Alfaro, J., Lafourcade, P. (eds.) FPS 2011: Proceedings of the 4th MITACS Workshop on Foundations & Practice of Security, LNCS, vol. 6888, pp. 208–217. Springer (2011)
30. Papagiannakopoulou, E.I., Koukovini, M.N., Lioudakis, G.V., Garcia-Alfaro, J., Kaklamani, D.I., Venieris, I.S., Cuppens, F., Cuppens-Boulahia, N.: A privacy-aware access control model for distributed network monitoring. Computers and Electrical Engineering (2012)
31. Preda, S., Cuppens, F., Cuppens-Boulahia, N., Garcia-Alfaro, J., Toutain, L.: Dynamic deployment of context-aware access control policies for constrained security devices. Journal of Systems and Software 84, 1144–1159 (July 2011)
32. Priebe, T., Dobmeier, W., Kamprath, N.: Supporting attribute-based access control with ontologies. In: ARES 2006: Proceedings of the The First International Conference on Availability, Reliability and Security. pp. 465–472. IEEE Computer Society (2006)
33. Ravari, A., Amini, M., Jalili, R., Jafarian, J.: A history based semantic aware access control model using logical time. In: Computer and Information Technology, 2008. ICCIT 2008. 11th International Conference on. pp. 43–50 (December 2008)
34. Rota, A., Short, S., Rahaman, M.A.: Xml secure views using semantic access control. In: Proceedings of the 2010 EDBT/ICDT Workshops. pp. 5:1–5:10. EDBT '10, ACM, New York, NY, USA (2010)
35. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Proceedings of the 24th annual international conference on Theory and Applications of Cryptographic Techniques. pp. 457–473. EUROCRYPT'05, Springer-Verlag, Berlin, Heidelberg (2005)
36. Sandhu, R., Coyne, E., Feinstein, H., Youman, C.: Role-based access control models. IEEE Computer 29(2) (February 1996)
37. Solove, D.J.: A brief history of information privacy law. In: Wolf, C. (ed.) Proskauer on Privacy: A Guide to Privacy and Data Security Law in the Information Age, chap. 1, pp. 1–46. Practising Law Institute, New York, NY, USA (2006)
38. Sun, Y., Pan, P., Leung, H.f., Shi, B.: Ontology based hybrid access control for automatic interoperation. In: Xiao, B., Yang, L., Ma, J., Muller-Schloer, C., Hua, Y. (eds.) Autonomic and Trusted Computing, LNCS, vol. 4610, pp. 323–332. Springer Berlin / Heidelberg (2007)
39. The World Wide Web Consortium (W3C): OWL Web Ontology Language Overview (February 2004), W3C Recommendation
40. The World Wide Web Consortium (W3C): Web Services Description Language (WSDL) Version 2.0 (June 2007), W3C Standard
41. Tonti, G., Bradshaw, J., Jeffers, R., Montanari, R., Suri, N., Uszok, A.: Semantic web languages for policy representation and reasoning: A comparison of kaos, rei, and ponder. In: The SemanticWeb - ISWC 2003, LNCS, vol. 2870, pp. 419–437. Springer Berlin / Heidelberg (2003)
42. Yuan, E., Tong, J.: Attributed based access control (ABAC) for web services. In: ICWS '05: Proceedings of the IEEE International Conference on Web Services (2005)